8 Naive Bayes Classifier : sentiment analysis on a Twitter dataset

```
# Install required packages if not already installed
# pip install nltk scikit-learn pandas

import pandas as pd
import nltk
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.naive_bayes import MultinomialNB
from sklearn.metrics import classification_report, accuracy_score
# Download stopwords if not already present
ntk.download('stopwords')
from nltk.corpus import stopwords
import string

# sample Twitter dataset (replace with your own dataset for real
application)

 data = {
 'tweet': [
   "I love this phone!",
   "This movie is terrible..",
   "Had an awesome day today:)",
   "I hate waiting in traffic",
   "Such a boring game",
   "Best concert ever!",
"I'm so sad right now",
"What a great experience!",
 "Worst customer service",
"Feeling happy and blessed!"],
'Sentiment':[
 'Positive','negative','Positive','negative','negative','Positive',
 'negative','Positive','negative','Positive']}
#Load the dataset
 df = pd.DataFrame(data)
#Preprocessing function
def preprocess_text(text):
text = text.lower()
 text = ''.join([char for char in text if char not in
  string.punctuation])
tokens = text.split()
tokens = [word for word in tokens if word not in stopwords.
words('english')]
return ' '.join(tokens)

#Apply preprocessing
df['clean_tweet'] = df['tweet'].apply(preprocess_text)
```

```
#convert text to feature vectors
vectorizer = CountVectorizer()
X = vectorizer.fit_transform(df['clean_tweet'])
y = df['sentiment']

#split dataset
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size = 0.2, random_state = 42)

# Train Naive Bayes classifier
nb_classifier = MultinomialNB()
nb_classifier.fit(X_train, y_train)

#Make predictions
y_pred = nb_classifier.predict(X_test)

#Evaluate the model
print("Accuracy:", accuracy_score(y_test, y_pred))
print("\n classification Report:")
print(classification_report(y_test, y_pred))
```

8th Program

```
Accuracy: 0.0

Classification Report:
              precision    recall  f1-score   support

    negative       0.00      0.00      0.00       2.0
    positive       0.00      0.00      0.00       0.0

    accuracy                           0.00       2.0
   macro avg       0.00      0.00      0.00       2.0
weighted avg       0.00      0.00      0.00       2.0
```